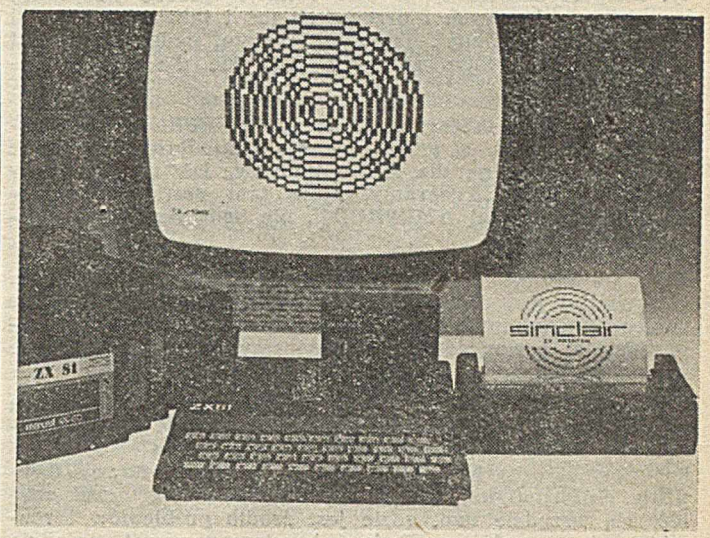
W czwartym mikroKLANIE przedstawiony został opis konstrukcji układu sterującego wyświetlaczami i umożliwiającego wprowadzenie danych z klawiatury do systemu mikroprocesorowego. Układ ten wymaga odpowiedniego oprogramowania. W tej części opisu przedstawiamy przykładowe procedury, które mogą posłużyć za wzorzec przy adaptowaniu rozwiązania na potrzeby konkretnego systemu. Procedury napisane zostały w języku ASSEMBLER 8080. Opisywane rozwiązanie zostało wykorzystane w systemie MSA80.1000, opracowanym w Przemysłowym Instytucie Elektroniki.

Klawiatura i wyświetlacze w 8-bitowym systemie mikroprocesorowym (2)

Do obsługi współpracy klawiatury i wyświetlaczy z systemem wykorzystane zostały procedury **GETKEY** i **DSPDGT.** Zmienne VARBS należy umieścić w pamięci RAM, natomiast pozostałą część programu można przechowywać w pamięci EPROM. Portom: wejściowemu, wyjściowemu i przełączającemu (symbole PINP, DISPLE, POUT) — należy przyporządkować rzeczywiste adresy.

Zastosowany w systemie układ zegara generuje co 2,5 ms przerwanie, w wyniku którego następuje wywołanie

```
; Procedura przetwarza cyfrę heksadecymalną podawaną w rejestrze A
; na kod sterujący wyświetlaczem 7 segmentowym.
; Rezultat zapisywany jest w buforze pod adresem
; wskazanym przez parę rejestrów DE.
; Zmienia zawartość rejestrów: H, L, D, E, A
DSPDGT: ANI     0FH     ; Wyzeruj nieistotne 4 starsze bity liczby re-
                        ; prezentującej cyfrę heksadecymalną
        LXI     H,CODGT ; Oblicz adres, pod którym umieszczono kod
        ADD     L       ; cyfry heksadecymalnej
        MOV     L,A     ;
        MOV     A,M     ; Pobierz kod cyfry
        STAX    D       ; Zapisz kody cyfry do odpowiedniego bufora
                        ; wyświetlacza
        INX     D       ; Zwiększ o 1 wskaźnik bufora wyświetlacza
        RET             ; Wróć
; Tablica kodów wyświetlacza cyfr heksadecymalnych
CODGT:  DB      3FH     ; 0
        DB      06H     ; 1
        DB      5BH     ; 2
        DB      4FH     ; 3
        DB      66H     ; 4
        DB      6DH     ; 5
        DB      7DH     ; 6
        DB      07H     ; 7
        DB      7FH     ; 8
        DB      6FH     ; 9
        DB      77H     ; A
        DB      7CH     ; B
        DB      39H     ; C
        DB      5EH     ; D
        DB      79H     ; E
        DB      71H     ; F
; Procedura wprowadzająca do akumulatora kod ASCII wciśniętego klawisza
; Jeżeli po poprzednim wywołaniu GETKEY nie wciśnięto żadnego klawisza, kod
; w akumulatorze=80H.
; Procedura zmienia zawartość rejestrów: H, L, A
GETKEY: LXI     H,KEYBUF ; Załaduj adres bufora
        MOV     A,M     ; Pobierz kod znaku z bufora
        MVI     M,80H   ; Zapisz kod 80H do bufora
        RET             ; Wróć
```



Przezorni na rynek

Rozmowa z przedstawicielami Przedsiębiorstwa Zagranicznego AMEPROD — WOJCIECHEM LIPKO, dyrektorem ds. systemów komputerowych, i GRZEGORZEM KORYTOWSKIM, z-cą dyrektora ds. elektroniki

µK: Bez wielkiej chyba przesady można powiedzieć, że AMEPROD wprowadził mikroinformatykę na polski rynek. Od kiedy ZX81 można oficjalnie kupić za złotówki, przestała być ona tylko wspaniałą wizją dalekiej przyszłości, staje się faktem. Czy zdawaliście sobie Panowie sprawę z istoty swego przedsięwzięcia, czy też była to po prostu kolejna transakcja?

Grzegorz Korytowski: — Zaczęło się od czysto handlowej analizy. Mieliśmy dewizy z eksportu innych wyrobów, chcieliśmy więc rozpocząć taką produkcję — w ramach elektroniki — w której bylibyśmy na polskim rynku pionierami, a która zapewniała powodzenie. Stąd mikrokomputery. Choć oczywiście, w tej głębi kryzysu, przy ogromnych brakach na każdym kroku, można się było zająć czymkolwiek, np. długopisami z zegarkiem...

µK: — Kiedy to było?

G.K.: — Na Międzynarodowych Targach Poznańskich w 1982 roku. Zorientowaliśmy się wówczas, jakie są nasze możliwości eksportowe. Poza tym musieliśmy wspomóc dewizami produkcję na krajowy rynek — tak, aby były złotówki na produkcję eksportową... Wybór mikrokomputerów był następstwem moich zainteresowań hobbystycznych, może nawet jakiegoś fanatyzmu. Ale gdy w październiku 1982 sprowadziliśmy pierwsze egzemplarze ZX81, nie bardzo wierzyłem w sukces. To co się stało, było zresztą trudne do przewidzenia. Nie była potrzebna żadna reklama, żaden marketing. W ciągu kilku miesięcy teka zamówień przerosła możliwości produkcyjne. I w efekcie sens społeczny naszego przedsięwzięcia stał się oczywisty.

µK: — Wróćmy na chwilę do początków. Od kiedy AMEPROD istnieje?

G.K.: — Od kwietnia 1981. Uruchomiliśmy najpierw zakład metalowy, który wykonywał meble ogrodowe, sanki, stoliki, a także zakład elektroniki, produkujący dzwonki

Przezorni i rynek

elektryczne, gry telewizyjne... wszystko — proste przedmioty. Eksport zaczęliśmy w 1982 roku — przede wszystkim wyrobów metalowych i drzewnych (bo i taki zakład został wkrótce uruchomiony). Trwa on, zresztą, nadal. Natomiast największym źródłem złotówek stała się dla nas mikroinformatyka.

µK: — Zastanawialiśmy się dlaczego AMEPROD, nie organizacja społeczna, a przedsiębiorstwo handlowe, inwestuje w tak nieopłacalne dla siebie imprezy, jak np. konkursy klubu ABAKUS. Programy, które tam powstaną, nagrodzone komputerami AMEPRODU, nie będą nawet własnością firmy...

G.K.: — To nie jest działalność wyłącznie charytatywna. Oczywiście, jest spory element ryzyka, ale przecież takie społeczne działania są w stanie wylansować świetnych programistów, przynieść dobre programy. A na tym także można zarabiać.

Wojciech Lipko: — Nie można zajmować się informatyką bez zrozumienia głębszego sensu tej dziedziny. Już dotychczasowe nasze doświadczenie pozwala pojąć wielkość „mikroinformatycznych" przeobrażeń, znaczenie nowej dziedziny. Informatyka, dzięki mikrokomputerom, może znaleźć się wszędzie tam, gdzie jest źródło problemów i rozwiązać je na miejscu. Duży i kosztowny sprzęt staje się teraz poręcznym narzędziem, mogącym istotnie zmieniać nasze życie. Jest to też forma politechnizacji społeczeństwa, koniecznej, zważywszy nadchodzące zmiany. Musimy patrzeć perspektywicznie.

ZX81 jest dzisiaj najpopularniejszym komputerem w Polsce, biorąc pod uwagę liczbę konfiguracji. Jest ich ponad 600. Użytkownicy tych komputerów, niezależnie — od nas kupionych czy przywiezionych z zagranicy — są potencjalnymi nabywcami naszego oprogramowania, którym także zaczynamy handlować. Jest to komputer pierwszego kroku, od którego można rozpocząć kontakt z informatyką, by potem przejść do poważniejszych, profesjonalnych zastosowań, do większych systemów. Może być on podstawą edukacji szkolnej i uczelnianej, właśnie na prawach elementarza.

Mikrokomputera o tych rozmiarach i takich parametrach nie można dzisiaj robić w kraju bez znacznego udziału dewiz. Przemysł kluczowy, ze swoim samofinansowaniem, nie jest w stanie tego dokonać. My mamy stosunkowo większe możliwości, więc jest to też jakby nasz obowiązek.

G.K.: — Tak jak w innych przedsiębiorstwach zagranicznych, i u nas dominuje biznes. Ale my, ponadto, jesteśmy mikroinformatycznymi zapaleńcami. To w istocie sprawa satysfakcji.

W.L.: — Plany wielu przedsiębiorstw, w tym naszego, wynikają z kultury technicznej ludzi, którzy je tworzą. Rozstrzyga środowisko.

µK: — Jakie więc plany?

W.L.: — Poza samymi mikrokomputerami, chcemy się zająć oprogramowaniem. Pod koniec ubiegłego roku powstał, pod naszym patronatem, klub użytkowników ZX81. Mamy ponad sto zgłoszeń różnych instytucji i prywatnych właścicieli komputerów. Na pierwszym spotkaniu było ponad 250 osób. Mając rozeznanie w oprogramowaniu powstałym u użytkowników, będziemy nabywać najciekawsze, weryfikować je, uzupełniać opisy i przekazywać innym chętnym. Powołaliśmy też redakcję biuletynu użytkowników — informatora techniczno-handlowego. Przygotowujemy do druku już drugi numer.

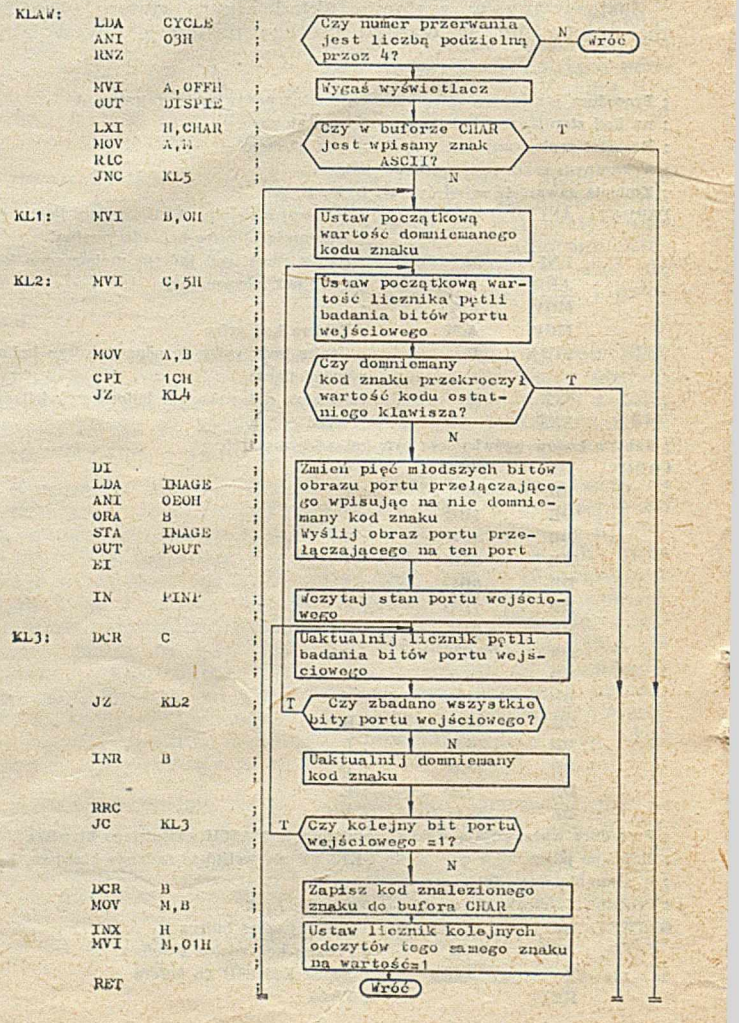
µK: — Wszystko to — za pieniądze?

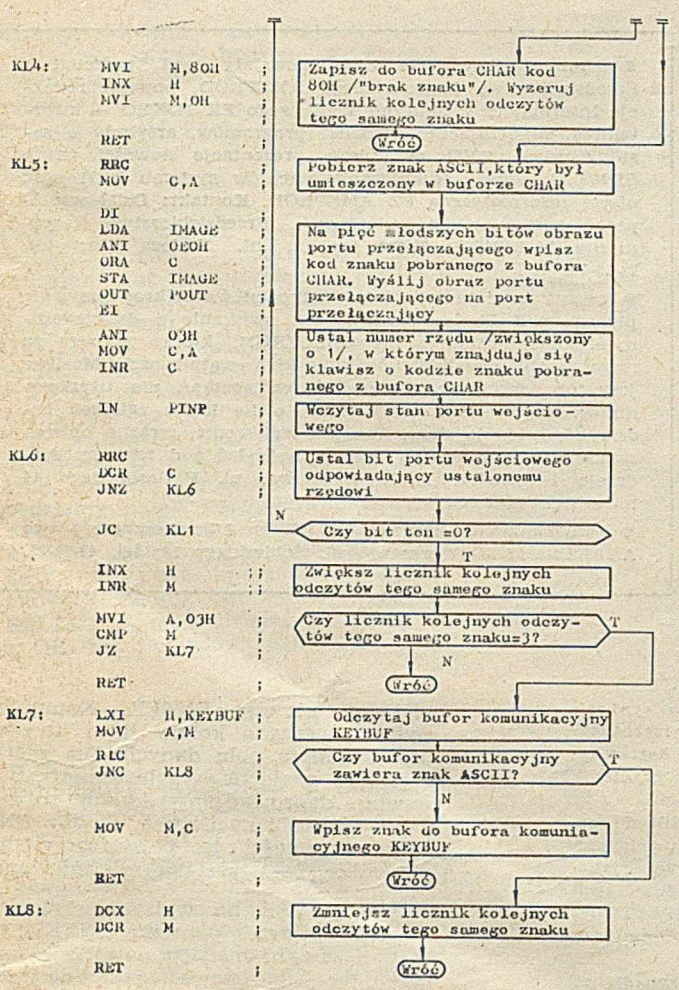
W.L.: — Oczywiście. Nas nikt nie dotuje, musimy się sami utrzymać. Odsprzedając państwu tyle, ile żąda dolarów, płacąc podatki...

```
; Segment zmiennych programu obslugi klawiatury i wyświetlaczy
VARBS:  IMAGE:   DS 1     ; Obraz portu przełączającego
        CHAR:    DS 1     ; Bufor dekodowanego znaku
        COUNT:   DS 1     ; Licznik klawiatury — umożliwia uniezależ-
                          ; nienie się od drgań zestyków
        KEYBUF:  DS 1     ; Bufor komunikacyjny znaku klawiatury
        DISPLE:  DS 8     ; 8 buforów wyświetlaczy
        CYCLE:   DS 2     ; Licznik przerwań (modulo 10000H)

; Procedura obslugi wyświetlaczy
WYSW:   MVI   A,0FFH      ; Wygaś wyświetlacze
        OUT   DISPL
        LDA   CYCLE       ; Pobierz numer (3 bitowy — 0 do 7)
        ANI   07H         ; sterowanego wyświetlacza
        MVI   H,0         ; Przygotuj adres bufora sterowanego wyświe-
        MOV   L,A         ; tlacza
        DAD   D.
        RLC               ; Przygotuj numer wyświetlacza do podania
        RLC               ; na bity 4, 3, 2 portu przełączającego
        MOV   C,A
        DI                ; Bity 7, 6, 5, obrazu portu przełączającego
        LDA   IMAGE       ; pozostaw bez zmiany. Na bitach 4, 3, 2
        ANI   0E0H        ; zestaw numer wyświetlacza. Bity 1, 0 wy-
        ORA   C           ; zeruj
        STA   IMAGE
        OUT   POUT        ; Wyślij obraz portu przełączającego na port
        EI                ; przełączający
        MOV   A,M         ; Pobierz z bufora sterowanego wyświetlacza
        CMA               ; kod znaku
        OUT   DISPL       ; Wylij go na port wyjściowy
        RET               ; Wróć

; Procedura obsługi klawiatury
KLAW:   LDA   CYCLE       ;
        ANI   03H         ;
        RNZ               ;
        MVI   A,0FFH      ;
        OUT   DISPLE      ;
        LXI   H,CHAR      ;
        MOV   A,M         ;
        RLC               ;
        JNC   KL5         ;
KL1:    MVI   B,0H        ;
KL2:    MVI   C,5H        ;
        MOV   A,B         ;
        CPI   1CH         ;
        JZ    KL4         ;
        DI                ;
        LDA   IMAGE       ;
        ANI   0E0H        ;
        ORA   B           ;
        STA   IMAGE       ;
        OUT   POUT        ;
        EI                ;
        IN    PINP        ;
KL3:    DCR   C           ;
        JZ    KL2         ;
        INR   B           ;
        RRC               ;
        JC    KL3         ;
        DCR   B           ;
        MOV   M,B         ;
        INX   H           ;
        MVI   M,01H       ;
        RET               ;
```





procedury obsługującej wyświetlacze **WYSW** i klawiaturę **KLAW** oraz zwiększenie o 1 licznika przerwań **CYCLE** (zliczającego modulo 10000H).

Dla każdego z ośmiu wyświetlaczy przewidziano 1-bajtowy bufor. Wywołanie procedury **DSPDGT** powoduje wpisanie kodu do odpowiedniego bufora. Natomiast wywoływana przez przerwanie procedura **WYSW** powoduje wysyłanie przez 2,5 ms na linie 4, 3, 2 portu POUT numeru wyświetlacza (przy każdym przerwaniu innego), a na port DISPLE — kodu wyświetlanej cyfry, pobranego z odpowiedniego bufora.

W systemie zastosowano matrycę złożoną z 26 klawiszy (w tym 16 klawiszy do wprowadzania liczb w kodzie szesnastkowym), rozmieszczonych w 7 kolumnach (numerowane od 0 do 6). W kolumnach 5 i 6 umieszczono po trzy klawisze, a w pozostałych po cztery. Cyfry heksadecymalne zajmują kolumny — 0 do 3.

Procedura **KLAW** realizowana jest raz na cztery przerwania. Sprawdza ona, czy został wciśnięty klawisz — poprzez wygaszenie wyświetlaczy, wybieranie kolejnych kolumn matrycy i poszukiwanie „zera" wśród bitów portu wejściowego PINP. W kodzie każdego klawisza bity 0, 1 są numerem rzędu, zaś 2, 3, 4 numerem kolumny matrycy klawiatury. Kod wciśniętego klawisza wpisywany jest do bufora CHAR. Jeśli trzykrotnie pod rząd zostanie wykryte wciśnięcie tego samego klawisza, kod znaku zostaje umieszczony w drugim buforze (komunikacyjnym) KEYBUF, skąd można go odczytać za pomocą procedury **GETKEY.**

ZBIGNIEW POJMAŃSKI
PIE, Warszawa

Przezorni na rynek

µK: — Wróćmy do planów.

G.K.: — Pakiet zamówień na ZX81 na rok bieżący został już zamknięty, a nie przewidujemy kontynuacji tej produkcji w roku 1985. Najprawdopodobniej baza elementowa będzie już na Zachodzie niedostępna, w efekcie rezygnacji firmy SINCLAIR z tego modelu. Będziemy się natomiast zajmować rozwojem konfiguracji, nowymi jej elementami (np. drukarka SEIKOSHA dla zastosowań profesjonalnych, czy rozbudowa pamięci do 64 K).

W.L.: — Przyjęte zamówienia zrealizujemy. Cały też rozwój konfiguracji, oprogramowania — będzie dokonywany w oparciu o własne rozwiązania. Nie zostawimy klientów samych sobie. Nie zostaną bez opieki — to pragnąłbym podkreślić.

G.K.: — Na tegorocznych Międzynarodowych Targach w Poznaniu zamierzamy zaprezentować nowy mikrokomputer o lepszych parametrach, bardziej uniwersalny od ZX81, częściowo wykorzystujący krajowe elementy. Nie chcemy na razie ujawniać szczegółów. Ważne jednak, że zostajemy przy sprzęcie dla amatorów, początkujących, nie podejmując konkurencji w produkcji systemów profesjonalnych. Chcemy umożliwić użytkownikom kontakt z komputerem przy możliwie najniższych nakładach finansowych, przy użyciu telewizora, magnetofonu kasetowego. Różnimy się więc w założeniach od innych firm, z którymi jesteśmy czasem porównywani — IMPOL, COMPUTEX czy ELWRO. My także chcemy sprzętu efektywnego, lecz — najtańszego!

µK: — Właśnie, pomówmy o cenach. Często się słyszy, że za ZX81 AMEPROD bierze za dużo...

W.L.: — My nie jesteśmy tylko przewoźnikiem i sklepem, ale zakładem produkcyjnym, który z elementów zachodnich tworzy komputer. Koszty wykonania są w kraju wysokie i musimy uwzględniać to w cenie produktu. Poza tym — jak już mówiłem — dolary sprzedawane państwu (50% obrotu) i podatek także robią swoje. Ponadto — utrzymanie grupy serwisowej; nie odsyłamy przecież użytkowników do Anglii.

ZX81 musi też zarobić na rozwój. Nie otrzymujemy przecież kredytów...

G.K.: — Udzielamy gwarancji. To kosztuje, bo ten sprzęt mimo wszystko nie jest pozbawiony wad. Tym bardziej, gdy pracuje intensywniej niż powinien w zastosowaniach amatorskich (np. drukarka), dla których jest stworzony. Ale z drugiej strony — okazuje się, że nawet osoby prywatne często chcą kupować u nas, a nie — co jest znacznie tańsze — sprowadzać z Zachodu.

µK: — Częste są przypadki sprzedaży ZX81 prywatnym osobom?

G.K.: — Bardzo rzadkie, chociaż w zamówieniach jest to pół na pół. Musimy działać strategicznie. Z wielu względów bardziej nam się opłaca sprzedawać komputery instytucjom. Taka działalność, jak nasza, nie jest przez władze doceniana. Często robi nam się różne trudności, a to z pozwoleniem na import, a to na eksport... Bo właściwie po co komu jakieś tam komputery. Wtedy my pokazujemy pisma, czy to z WAT, czy z jakiegoś ministerstwa, szkoły wyższej bądź instytutu, czyli dla władzy — pisma klientów poważnych, i wówczas przeszkodę łatwiej pokonać.

W.L.: — I w efekcie ZX81 jest u nas najczęściej stosowany w różnych instytucjach przez profesjonalistów. A do takich niezbyt się nadaje. Propozycja Sinclaira: taniość i prostota nade wszystko — zaczyna stanowić wadę przy intensywnej eksploatacji. Ustawiczny montaż i demontaż konfiguracji niszczy styki, które nie są dostatecznie trwałe. Musimy więc uwzględniać ten nasz rodzimy sposób funkcjonowania ZX81.

µK: — Wiele mówi się o tym, że niedawne pociągnięcia podatkowe fiskusa sparaliżują działalność przedsiębiorstw zagranicznych...

Przezorni nä rynek

W.L.: — Decyzje podatkowe z sierpnia ub.r. uderzyły w nasz rynek. Praca AMEPRODU jest nadal opłacalna, ale produkcja nie może być rozwijana ilościowo. Tak wielki procent dewiz (50% z obrotu, nie z zysku!), jaki musimy przekazywać państwu, może dla wielu przedsiębiorstw oznaczać koniec pracy. Wiele elementów niestety trzeba kupować za dolary. Jeśli stanowią one — powiedzmy — 60% obrotu a 50% trzeba oddać fiskusowi, to 10% musimy dołożyć...

µK: — Ale wśród Panów raczej optymizm!?

G.K.: — Mamy już sporo doświadczenia, pewien staż — to ułatwia potyczki. W 1983 roku nasz eksport wyniósł — na 500 istniejących w kraju firm zagranicznych — 8%. Mamy nadzieję, że działając przezornie przetrwamy ciężkie czasy. Niemniej wprowadzone przepisy znacznie ograniczą produkcję, to oczywiste.

W.L.: — I ostatecznie — zmniejszą ilość dewiz wpływających do kasy państwa. Najbardziej zaś ucierpi krajowy rynek.

Z. GLUZA
Oprac.

- Klub użytkowników ZX81, powstały pod patronatem Przedsiębiorstwa Zagranicznego AMEPROD, powołał redakcję informatora techniczno-handlowego ZX MIKRO — kwartalnika, zawierającego wydruki programów, artykuły o zastosowaniach ZX81 w Polsce, prezentacje nowości firmy SINCLAIR, opisy techniczne elementów systemu ZX81 oraz ofertę informatyczną PZ AMEPROD. Kontakt: Dział współpracy systemów komputerowych, Przedsiębiorstwo Zagraniczne AMEPROD; 61-632 Poznań, ul. Kmieca 20a; tel. 22-18-79.

- Uwaga nowicjusze! Klub Użytkowników Mikroprocesorów planuje pod koniec września br. spotkanie z doświadczonymi użytkownikami ZX SPECTRUM, którzy pomogą w rozproszeniu wszelkich debiutanckich wątpliwości. Możliwe jest też zorganizowanie podobnych spotkań dla użytkowników innych mikrokomputerów, o ile liczba zgłoszeń będzie odpowiednio duża. Wstępne zgłoszenia, wraz z opisem zasadniczych problemów, należy nadsyłać pod adresem Andrzeja Drożniaka: 00-777 Warszawa, ul. Pytlasińskiego 14 m. 4 (tel. 41-26-01).

- Przypominamy kontakty do Klubu Mikrokomputerowego ABAKUS: 02-930 Warszawa, ul. Sobieskiego 18; tel. 42-91-85.

Po półtorarocznej karierze ZX SPECTRUM (p. mikroKLAN 4), na rynkach zachodnich oferowany jest olbrzymi wybór (kilkaset) programów przeznaczonych dla tego mikrokomputera. Znakomita większość z nich to gry, dostarczające rozrywki, co może utwierdzać w przekonaniu, że mamy do czynienia z bardzo inteligentną, ale jednak — zabawką.

Czarnorynkowy kurs twardych walut zmusza nas byśmy podchodzili do zabawek z uwagą godną profesjonalnego sprzętu komputerowego. Użytkownicy na Zachodzie, płacąc zaledwie kilkaset dolarów więcej, mogą nabyć sprzęt znacznie bardziej efektywny; krajowych użytkowników zazwyczaj nie stać na tych kilkaset dolarów. A niedogodności związane z wykorzystaniem ZX SPECTRUM są mimo wszystko znacznie mniejsze niż trudności, przed jakimi stoimy przy braku mikrokomputerów.

Rozpoczynając serię publikacji o programach użytkowych, nieprzypadkowo wybraliśmy jako pierwszy program VU-File. Oprogramowanie typu baza danych jest u nas realizowane — jak dotąd — głównie na dużych systemach. Natomiast VU File nie jest przeznaczone dla Elektronicznych Ośrodków Obliczeniowych, lecz dla użytkowników ZX SPECTRUM. Ma więc szansę trafić pod strzechy. Program pozwala zaoszczędzić wiele czasu traconego na żmudne sporządzanie rozmaitych zestawień i prowadzenie ewidencji — czynności, które tak kocha nasza rodzima biurokracja.

Program katalogujący VU-File (ZX SPECTRUM)

Program ten jest ulepszoną wersją analogicznego programu dla ZX 81. Wszystkie istotne części programu napisane są w języku ASSEMBLER Z80. Program działa więc bardzo szybko i zajmuje tylko 5 KB pamięci.

Program składa się z dwóch części. W pierwszej projektuje się karty katalogu i zapełnia je danymi. Druga umożliwia różnorodne wykorzystanie przygotowywanych uprzednio kart.

Zakładanie katalogu

Użytkownik ma do dyspozycji czystą kartę, na której w dowolnych miejscach, za pomocą edytora ekranowego umieszcza teksty nagłówków. Edytor ekranowy bardzo ułatwia projektowanie kart. Przykładowo — karta z nagłówkiem wyglądać może następująco:

Nazwisko
Imię
Branża materiałowa
Towary

Adres

Po wpisaniu wszystkich nagłówków przyciskamy równocześnie klawisze SYMBOL SHIFT i STOP. Wtedy na ekranie pojawia się nowy typ kursora i można przejść do projektowania pól danych. W tym celu doprowadzamy kursor do pozycji, od której chcemy, aby zaczynało się pole danych i naciskamy klawisz ENTER. Komputer pyta wówczas o kolory tła i liter w definiowanym polu danych. Dla zdefiniowania następnego pola danych, ponownie doprowadzamy kursor w żądane miejsce i naciskamy klawisz ENTER. Czynności te powtarzamy, aż wszystkie pola zostaną zdefiniowane.

Uwaga: jeśli po wpisaniu kolorów naciśniemy przez pomyłkę ENTER, to w tym samym miejscu powstanie nakładające się „nowe" pole danych. Niestety, nie można go skasować i trzeba zaczynać wszystko od początku. Ten fragment programu nie został odpowiednio dopracowany przez autorów.

Po opisaniu wszystkich pól danych, możemy przejść do wpisania w nie odpowiednich informacji. W tym celu przyciskamy równocześnie klawisze SYMBOL SHIFT i STOP. W odpowiedzi kursor ustawia się w odpowiednim polu i możemy wpisywać dane.

Niestety, jedna linia danych nie może być kontynuowana w następnej linii na ekranie. Jest to kolejna niedogodność. Jeśli więc wiemy, że informacje mogą nie zmieścić się w jednym wierszu, musimy pod nagłówkiem zdefiniować kilka linii pola danych.

Po zapisaniu karty możemy zapisać następne lub nacisnąć SYMBOL SHIFT i STOP, co spowoduje przejście do programu umożliwiającego korzystanie z katalogu.

Korzystanie z katalogu

W tej części programu przewidziano następujące polecenia:

ENTER — wprowadza do części programu, w której wpisujemy nowe karty.

ALTER — służy do wprowadzenia zmian na karcie będącej w danym momencie na ekranie.

COPY — powoduje przepisanie na drukarkę całej karty będącej aktualnie na ekranie.

DELETE — kasuje kartę znajdującą się na ekranie.

PRINT — włącza specjalną opcję oszczędnościową. Użytkownik może zaprojektować rozmieszczenie nagłówków i pól danych niezależnie dla drukarki i ekranu.

ORDER — wskazuje komputerowi, według którego pola mają być uporządkowane karty. Kolejność sprawdzana jest na podstawie kodu ASCII. Gdy wpisujemy liczby, kolejne cyfry dziesiętne powinny znajdować się w tych samych miejscach. Dzięki temu będziemy mogli porządkować również według wielkości liczbowych. Przykładem zastosowania polecenia ORDER może być uporządkowanie dostawców według cen ich wyrobów.

SELECT — może działać równocześnie z ORDER. Powoduje wybieranie przez komputer tylko takich kart, na których występuje określony ciąg znaków w danym polu (bądź na całej karcie). W połączeniu z poprzednim poleceniem można — na przykład — uporządkować według ceny dostawców tylko z Warszawy.

LIST — umożliwia automatyczne przeglądanie kart z szybkością jednej karty na sekundę. Wynik można również skierować na drukarkę.

INFORM — podaje informację o wielkości zajętego obszaru pamięci.

QUIT — umożliwia przejście do programu, w którym mamy następujące opcje:

SAVE file — nagranie całego katalogu na taśmę,

LOAD file — wprowadzenie całego katalogu z taśmy

ERASE — czyli kasowanie katalogu w pamięci komputera.

Program bardzo oszczędnie wykorzystuje pamięć. Na przykład, katalog 515 państw świata, gdzie o każdym są następujące dane: nazwa, stolica, waluta, język, dochód narodowy, liczba mieszkańców, powierzchnia, kontynent — zajmuje ok. 10 KB pamięci.

Wśród przykładów zastosowania programu VU-File, warto wymienić katalogi:

— książek upchniętych na najwyższą półkę
— programów (komputerowych bądź video)
— monet lub znaczków (a jak ktoś ma, to i obrazów Picassa)
— układów scalonych (niestety bez schematów)
— znajomych, których trudno spamiętać (z uwzględnieniem płci bądź innych walorów)
— rzeczy wypożyczonych.

Program ten powinien znaleźć się w bibliotece każdego posiadacza ZX SPECTRUM. Może on posłużyć do walki z tymi siłami domowymi, które twierdzą, że komputer to tylko droga zabawka.

PIOTR PARLEWICZ
Warszawa

Procedury mnożenia i dzielenia dla 8080

Wykonywanie jakichkolwiek obliczeń za pomocą mikrokomputera łączy się zazwyczaj z koniecznością mnożenia i dzielenia. Przedstawione poniżej procedury charakteryzują się stosunkowo krótkim czasem realizacji, co jest szczególnie ważne przy bardziej złożonych obliczeniach. Dla wykonania dzielenia potrzeba w najgorszym przypadku 1745 okresów zegarowych, a dla mnożenia 1023 okresy. Przy najczęściej stosowanej częstotliwości zegara 2,048 MHz daje to dla dzielenia ok. 852 μs, a dla mnożenia — ok. 500 μs.

Dzielenie

Procedura realizuje dzielenie liczby 32-bitowej (rejestry HL i DE — bajt najbardziej znaczący, MSB w rejestrze H) przez liczbę 16-bitową (para rejestrów BC). Powstały w wyniku dzielenia iloraz umieszczany jest w parze rejestrów DE, a reszta z dzielenia w parze rejestrów HL. Jeżeli iloraz przekroczy 16-bitowy zakres, następuje wyjście z procedury z sygnalizacją błędu (wyzerowany bit przeniesienia w rejestrze wskaźników). Oto program źródłowy w języku ASSEMBLER 8080:

```
DIV:    MOV  A,L      ; Czy przekroczony zakres?
        SUB  C
        MOV  A,H
        SBB  B
        RNC           ; Powrót jeśli tak
        MOV  A,B      ; Konwersja zawartości
        CMA           ; pary rejestrów BC
        MOV  B,A      ; na uzupełnienie dwójkowe
        MOV  A,C
        CMA
        MOV  C,A
        INX  B
        CALL LOOP     ; Dzielenie
        RET
; Podprogram 'LOOP' wykonuje dzielenie trzech bardziej znaczących bajtów
; dzielnej przez dwubajtowy dzielnik
LOOP:   MOV  A,D      ; Trzeci dzielony bajt do rej. A
        MOV  D,E      ; Zachowanie najmniej znaczącego bajtu dzielnej
                      ; lub najbardziej znaczącego bajtu ilorazu
        MVI  E,8      ; Licznik pętli 1
LOOP1:  DAD  H        ; Przesunięcie dzielnej w lewo
        JC   OVER     ; Rejestry HL przepełnione?
        ADD  A
        JNC  SUB
        INX  H        ; Dodanie przeniesienia jeśli wystąpiło
SUB:    PUSH H        ; Zachowanie dwóch bajtów dzielnej
        DAD  B        ; Odjęcie dzielnika
        JC   OK       ; Skok gdy wynik > 0
        POP  H        ; Odtworzenie dzielnej
        DCR  E        ; Dekrementacja licznika pętli 1
        JNZ  LOOP1
        MOV  E,A      ; Bajt ilorazu do rej. E
        STC
        RET
OK:     INX  SP       ; Eliminacja niepotrzebnej danej
        INX  SP
        INR  A        ; Wstawienie 1 do ilorazu
        DCR  E        ; Dekrementacja licznika pętli 1
        JNZ  LOOP1
        MOV  E,A      ; Bajt ilorazu do rej. E
        STC
        RET
OVER:   ADC  A        ; Dokończenie przesunięcia dzielnej,
                      ; wstawienie 1 do ilorazu
        JNC  OVRSUB
        INX  H        ; Dodanie przeniesienia jeśli jest
OVRSUB: DAD  B        ; Odjęcie dzielnika
        DCR  E        ; Dekrementacja licznika pętli 1
        JNZ  LOOP1
        MOV  E,A      ; Bajt ilorazu do rej. E
        STC
        RET
```

Mnożenie

Procedura realizuje mnożenie liczby 16-bitowej (para rejestrów BC) przez liczbę 16-bitową (para rejestrów DE). Iloczyn jest umieszczany w rejestrach DE-HL, przy czym najbardziej znaczący bajt wyniku w rejestrze D. W procedurze wykorzystywany jest podprogram wykonujący mnożenie liczby jednobajtowej przez dwubajtową („BMULT"). Liczba 8-bitowa umieszczana jest w akumu-

latorze A, a liczba 16-bitowa w parze rejestrów BC. Wynik mnożenia przekazywany jest przez podprogram w A i parze rejestrów HL (w akumulatorze jest najbardziej znaczący bajt wyniku). Czas realizacji podprogramu wynosi dla najbardziej niekorzystnego przypadku 424 cykle zegarowe.

Program źródłowy procedury mnożenia jest następujący:

```
MULT:   MOV   A,E     ; Mniej znaczący bajt mnożnika do A
        PUSH  D       ; Bardziej znaczący bajt na stos
        CALL  BMULT   ; Mnożenie cząstkowe
        XTHL          ; Wymiana mniej znaczących bajtów
        PUSH  PSW     ; Najbardziej znaczący bajt na stos
        MOV   A,H     ; Bardziej znaczący bajt mnożnika do A
        CALL  BMULT   ; Mnożenie cząstkowe
        MOV   D,A     ; Najbardziej znaczący bajt wyniku do D
        POP   PSW     ; Odtworzenie najbardziej znaczącego bajtu z 1
                      ; mnożenia
        ADD   H       ; Wyliczenie 3 bajtu wyniku
        MOV   E,A     ; i wprowadzenie go do rej. E
        JNC   NOCRR   ; Inkrementacja rej. D tylko gdy występuje prze-
        INR   D       ; niesienie
NOCRR:  MOV   H,L     ; Przesunięcie mniej znaczącego bajtu wyniku z 1
        MVI   L,0     ; mnożenia
        POP   B       ; Odtworzenie mniej znacz. bajtów 1 mnożenia
        DAD   B       ; Wyliczenie 2 mniej znacz. bajtów wyniku
        RNC           ; Koniec gdy nie ma przeniesienia
        INX   D       ; Korekcja 2 bardziej znacz. bajtów wyniku
        RET
; "BMULT" wykonuje mnożenie liczby 1-bajtowej przez 2-bajtową
BMULT:  LXI   H,0     ; Wstępne zerowanie wyniku
        LXI   D,7     ; Zerowanie D; rej. E — licznik
        ADD   A       ; Wydzielenie bitu mnożenia
LOOP1:  JNC   ZERO    ; Skok gdy równy zero
        DAD   B       ; Dodanie mnożnej
        ADC   D       ; Dodanie przeniesienia do 3 bajtu wyniku
ZERO:   DAD   H       ; Przesunięcie wyniku
        ADC   A
        DCR   E       ; Dekrementacja licznika
        JNZ   LOOP1
        RNC           ; Koniec gdy nie ma przeniesienia
        DAD   B       ; Gdy jest, ostatnie dodawanie
        ADC   D
        RET
```

Opracował AJP
na podstawie ELECTRONICS
z 24 lutego 1982

SINCLAIR rezygnuje już z produkcji ZX81. W ślad za nim rezygnację ogłasza rodzimy AMEPROD. Czy zatem wyszukiwanie w przestarzałym rozwiązaniu niuansów, które pozwolą coś tam rozwiązać efektywniej ma sens? Niestety, ma. Sytuacja rynkowa — jak wróżka z bajki — sprawia, że używane w kraju komputery (te mikro i te makro) są wiecznie młode. A przynajmniej muszą w to wierzyć ich użytkownicy, bo inaczej dla zachowania zdrowia psychicznego musieliby przerzucić się na produkcję guzików.

Tak więc życie zmusza nas do maksymalnego wykorzystania posiadanego sprzętu. To prawda, że niektóre rozwiązania ocierają się o granice zdrowego rozsądku; często jednak lepiej rozwiązać jakiś problem przez „postawienie na głowie", niż nie rozwiązać w ogóle. Z drugiej strony — problemy, które na pierwszy rzut oka wymagają znacznie bardziej wyrafinowanego sprzętu — po zastosowaniu kilku trików można często zrealizować nawet za pomocą poczciwego ZX81.

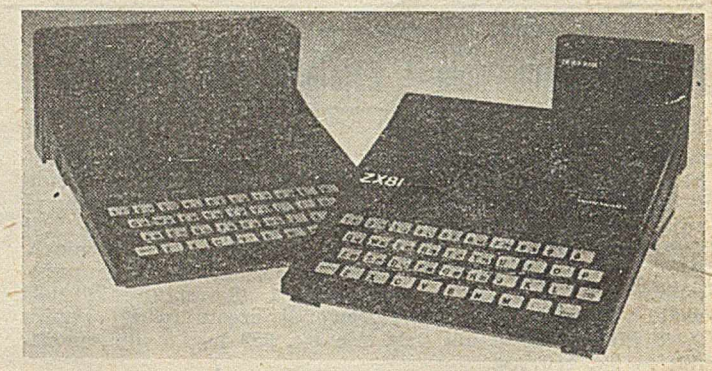
Połączenie dwóch ZX81

ZX81 może być traktowany jako moduł centralny, który został zbudowany w oparciu o najbardziej rozpowszechniony 8-bitowy mikroprocesor Z80. Względnie łatwe jest jego rozbudowywanie, programowanie w języku wewnętrznym, dopasowanie systemu do konkretnego zastosowania itp.

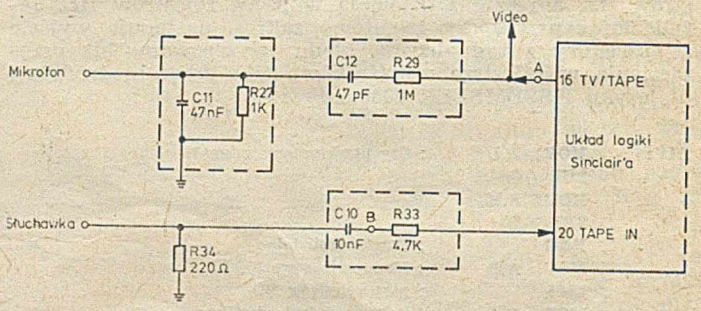
Proponowane rozwiązanie polega na wykonaniu połączeń między dwoma ZX81, z których jeden może być traktowany jako pamięć masowa dla drugiego. Możliwe jest również wiele innych zastosowań takiego połączenia: kontrola zapamiętania programu, przekazywanie danych z jednego programu do drugiego itd.

Zasada zapamiętywania programu na kasecie

Po podaniu instrukcji **SAVE**, układ logiczny ZX81 przechodzi w stan aktywny i jego wyjście TV/TAPE zmienia stan. Transmisja informacji wykonywania jest bit po bicie (ciąg impulsów o częstotliwości 3,3 kHz). Sygnał ten



nie może być podany bezpośrednio na wejście mikrofonowe magnetofonu, gdyż typowa jego czułość wynosi 5 mV. Amplituda sygnału wynosi 5 V, konieczne jest więc tysiąckrotne jej stłumienie.



Rys. 1. Filtry RC w ZX61 umożliwiające tłumienie i selekcję wykorzystywanego sygnału

ZX81 został wyposażony w filtr pasmowy (rys. 1) o częstotliwości środkowej równej 3,3 kHz. Składa się on z filtru górnoprzepustowego (szeregowy układ RC) oraz filtru dolnoprzepustowego (równoległy układ RC). Tłumienie filtru w paśmie przepustowym wynika z istnienia dzielnika tworzonego przez rezystancje R_{29} i R_{27}. Wynosi ono:

$$\frac{R_{27}}{R_{27}+R_{29}} = \frac{1}{1000}$$

jako że $R_{29} = 1000\ R_{27}$.

„Górna" częstotliwość graniczna może być wyznaczona jako:

$$F_g = \frac{1}{2\pi R_{29} C_{12}} = 3{,}38\ \text{kHz}$$

Podobnie „dolna" częstotliwość graniczna dana jest zależnością:

$$F_d = \frac{1}{2\pi R_{27} C_{11}} = 3{,}38\ \text{kHz}$$

Sygnał jest w ten sposób prawidłowo filtrowany, podczas gdy jego amplituda obniżona do 5 mV dopasowana jest do czułości wejściowej magnetofonu kasetowego.

Odtwarzanie zapisanego programu

Po podaniu instrukcji **LOAD** uaktywniony zostaje układ logiczny i wprowadzana jest informacja odpowiadająca sygnałowi wejściowemu. Sygnał ten przed dojściem do układu logicznego przechodzi przez filtr górnoprzepustowy. W stanie spoczynkowym (bez sygnału na wejściu) przez rezystancję R_{33} nie płynie prąd, gdyż kondensator C_{10} blokuje składową stałą. Wejście znajduje się „w powietrzu", tj. w stanie „1" w logice TTL.

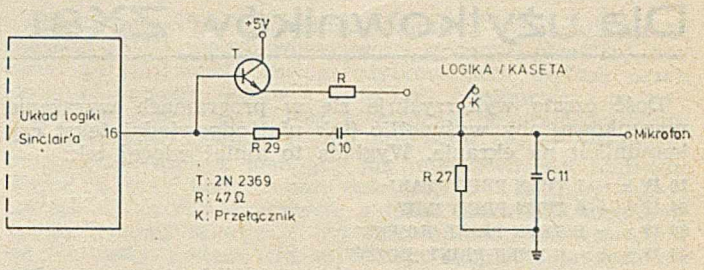
Przejście z „1" na „0" powoduje uaktywnienie wejścia układu logicznego. Konieczne jest zastosowanie sygnału o odpowiednim poziomie, a zmiany stanu powinny być bardzo szybkie. Jest to związane z pasmem transmitowanego sygnału: strome zbocza oznaczają szersze pasmo. Wskazane jest więc „podbicie" wysokich tonów w magnetofonie, w przeciwnym razie sygnał może być za słaby, a jego opadające zbocze nie wystarczająco strome, aby zmienić stan wejścia układu.

Należy przy tym uważać, aby nie podwyższyć za bardzo sygnału i nie wprowadzić zakłóceń lub zniekształceń, które mogłyby zmienić stan wejścia.

Połączenie dwóch ZX81

Wykonanie połączenia między dwoma ZX81 polega na zamontowaniu dodatkowego obwodu. Dla zapewnienia właściwego połączenia konieczne jest wzmocnienie wysyłanego sygnału (lub raczej nie tłumienie go). Obwód został zrealizowany przy użyciu tranzystora w układzie wtórnika emiterowego (wzmocnienie = 1, niska impedancja wyjściowa i wysoka impedancja wejściowa) z bazą podłączoną do wyjścia TV/TAPE.

Na rysunku 2 pokazano, że na wyjściu zastosowano rezystor 47 Ω, który ogranicza prąd płynący przez tranzystor do wartości 100 mA w przypadku zwarcia wyjścia. Zastosowany przełącznik ma dwie pozycje: jedną — dla połączenia logicznego silnych sygnałów, drugą — dla połączenia słabych sygnałów podczas nagrywania na kasetę.



Rys. 2. Przełącznik umożliwiający przesyłanie danych do magnetofonu kasetowego lub pamięci drugiego ZX81

Montaż

Tranzystor montuje się w pobliżu modulatora UHF. Bazę należy przylutować do pola znajdującego się między opornikiem R_{24} i otworem do mocowania obwodu drukowanego. Kolektor podłącza się bezpośrednio do zasilania modulatora UHF.

Przełącznik może być montowany z boku urządzenia, koło wyjścia typu „jack" sygnału wizyjnego. Wyjście przełącznika należy połączyć z opornikiem R_{27} (w pobliżu wyjścia „jack"). Opornik R podlutowuje się do przełącznika i wykonuje połączenie z emiterem tranzystora przy użyciu przewodu.

Prawidłowość działania połączenia można łatwo sprawdzić wykorzystując diodę LED, przymocowaną do wtyczki. Katoda powinna być podłączona do masy (0V), tj. do zewnętrznej metalowej części wtyczki. Anodę podłącza się do wewnętrznej części wtyczki.

Dioda powinna się świecić tylko wtedy, gdy przełącznik jest w pozycji LOGIKA. Po podaniu instrukcji SAVE „P" i nastawieniu przełącznika w pozycji LOGIKA, dioda powinna świecić się, gdy ekran jest ciemny. Podczas transmisji dioda świeci się „słabo".

Wykorzystanie drugiego ZX81

ZX81 będący „nadajnikiem" należy przełączyć w pozycję LOGIKA, a następnie wprowadzić instrukcję LOAD „P" w ZX81 spełniający rolę „odbiornika". Do ZX81 „nadajnika" wprowadzić program:

```
10 SAVE „P"
20 PRINT „DZIAŁA POPRAWNIE"
30 GO TO 20
```

i wcisnąć RUN.

Po wykonaniu transmisji oba mikrokomputery wyświetlają ten sam tekst, co oznacza, że zawierają ten sam program.

Opracował ANDRZEJ KONTKIEWICZ
na podstawie „MICRO-SYSTEMES",
marzec 1983

● Piotr Jastrzębski jest od niedawna posiadaczem mikrokomputera ATARI 600 XL. Chętnie nawiąże kontakt z innymi użytkownikami — dla wymiany doświadczeń. Kontakt: 81-365 Gdynia, ul. Abrahama 56 m. 2; tel. 20-26-77.

● Mieczysław Gołoś jest użytkownikiem mikrokomputera NEW BRAIN (model A). Dysponuje ASSEMBLEREM, FORTH'EM i wieloma informacjami technicznymi. Chętnie wymieni doświadczenia z innymi użytkownikami. Kontakt: 01-122 Warszawa, ul. Batalionu Zośka 4A m. 28; tel. p. 25-98-13, d. 36-43-86.

● W różnych ośrodkach powstają „agentury" KUM. Rozpoczynamy druk adresów osób, które będą dysponowały ankietami klubowymi, bieżącymi informacjami, itp. **KRAKÓW**: Waldemar Skorupa, 30-079, Kijowska 11 m. 31, tel. d. 37-10-25, p. 22-52-00; **POZNAŃ**: Piotr Torz, 60-621, Grudzieniec 117 m. 6, tel. p. 408-51; **SZCZECIN**: Andrzej Górecki, 71-175, Włodkowica 23 m. 3, tel. d. 737-17, p. 430-51 w. 49; **LUBLIN**: Kazimierz Rybka, 20-954, Budowlana 5, tel. p. 450-71 w. 653. (c.d.n.)

Informacje mikroKLANU prosimy przekazywać niezwłocznie pod adresem INFORMATYKI: 00-041 Warszawa, ul. Jasna 14/16 p. 244 (tel. 27-71-40). Czas płynie.

Prowadzi Andrzej J. Piotrowski (tel. p. 43-66-11 w. 42, d. 48-22-85)

Dla użytkowników ZX81

Dość często wykorzystuje się w programach wyrażenia warunkowe, aby wyświetlić (lub nie) odpowiedni tekst czy komunikat na ekranie. Wygląda to mniej więcej tak:

```
10 IF x = 1 THEN PRINT 'TAK'
20 IF x = 0 THEN PRINT 'NIE'
30 IF y = E THEN PRINT 'ROWNE'
40 IF y <> E THEN PRINT 'ROZNE'
```

To samo można zapisać w sposób może nieco mniej przejrzysty, ale zajmujący znacznie mniej miejsca w pamięci:

```
10 PRINT 'TAK' AND (x=1); 'NIE' AND (x=0),,
'ROWNE' AND (Y=E),, 'ROZNE' AND (Y<>E).
```

Uwaga: Nawet gdy trzeci warunek nie jest spełniony, interpretowany jest zapis dwóch przecinków.

Czasem zdarza się, że musimy przenumerować linie w pewnych blokach programu. Wiąże się to jednak ze skorygowaniem argumentów we wszystkich instrukcjach GOTO odnoszących się do przenumerowanego bloku. W bardziej rozbudowanych programach stwarza to możliwość wielu pomyłek. Można tego uniknąć stosując pewną regułę przy pisaniu programów. Poszczególnym częściom programu nadajemy nazwy wykorzystując instrukcję REM. Na początku programu umieszczamy deklaracje przypisujące wspomnianym nazwom numer odpowiadający linii umieszczonej bezpośrednio za linią zawierającą komentarz. Jako argument instrukcji GOTO podajemy zamiast numeru linii nazwę zmiennej. Po przenumerowaniu bloku programowego wystarczy teraz przejrzeć linie zawierające instrukcję REM i odpowiednio zaktualizować listę deklaracji. A więc koniec ze żmudnym wyszukiwaniem instrukcji GOTO!

Operatory warunkowe można w ZX81 traktować jako funkcje bolowskie, bowiem wynik sprawdzania warunku przyjmuje wartość 1 lub 0. Tak więc poprawny jest np. zapis: PRINT A = B. W tabeli zestawiono operatory warunkowe i odpowiadające im wyrażenia boolowskie.

$A < B \rightarrow \overline{A} \wedge B$	$A <> B \rightarrow A \,[+]\, B$
$A > B \rightarrow A \wedge \overline{B}$	$A = B \rightarrow \overline{A \,[+]\, B}$
$A <= B \rightarrow \overline{A} \vee B$	$A * B \rightarrow A \wedge B$
$A <= B \rightarrow \overline{A \vee B}$	$A + B \rightarrow A \vee B$

[+] — oznacza różnicę symetryczną (ang. EXOR)

Stosując wyrażenia warunkowe (co znakomicie upraszcza programy) można modelować kombinacyjne układy logiczne (tzn. sieć bramek logicznych bez sprzężeń zwrotnych). W zapisach należy jednak pamiętać, że standardowe operatory logiczne (AND, OR, NOT) mają niższy priorytet niż wyrażenia arytmetyczne i należy stosować nawiasy. Wykorzystanie wyrażenia arytmetycznego A+B ogranicza się do przypadków, gdy nie może zajść sytuacja, że równocześnie A = 1 i B = 1.

A oto przykład programu umożliwiającego drukowanie tabeli logicznych:

```
10 FOR A=0 TO 1
20 FOR B=0 TO 1
30 FOR C=0 TO 1
.
.
240 FOR Z=0 TO 1
250 PRINT A;B;...Z;'=' ; (BADANE WYRAŻENIE LOGICZNE)
260 NEXT Z
270 NEXT Y
.
.
.
500 NEXT A
```

I jeszcze przykład wyrażenia odpowiadającego 4-bitowemu generatorowi parzystości:

```
A<>(B<>(C<>D))
```

Rozszyfrowywanie barwnych pasków na opornikach nie jest zajęciem superzłożonym. Dla kilku rezystorów nie opłaca się nawet ładować programu. Jeśli jednak w ramach wiosennych porządków będziemy musieli posegregować całe pudło, to ZX81 służy pomocą. Zamieszczone poniżej programy pracują w wersji angielskojęzycznej, może jednak ktoś z Czytelników pokusi się o ich „przetłumaczenie"?

W poniższym programie wystarczy wprowadzić tylko kilka początkowych liter nazwy koloru, tak aby nie mógł on pomylić się z innym:

```
 10 LET C$='BLACK BROWN RED   ORANGEYELLOW
    GREEN BLUE  VIOLETGRAY  WHITE GOLD  SILVER'
 20 DIM A(3)
200 SCROLL
105 FOR J=1 TO 3
110 PRINT "BAND ";J;"? "
120 INPUT B$
130 PRINT B$
135 SCROLL
140 FOR K=1 TO 11
150 IF B$ =C$(K*6+1 TO K*6+LEN B$) THEN GOTO 180
160 NEXT K
170 PRINT '?'
171 SCROLL
175 GOTO 110
180 IF K=10 THEN LET K=-1
181 IF K=11 THEN LET K=-2
185 LET A(J)=K
190 NEXT J
195 SCROLL
200 PRINT 'R= ' ; (10*A(1)+A(2))*10**A(3) ; ' OHMS'
205 SCROLL
206 SCROLL
207 CLEAR
210 GOTO 10
```

Drugi program wypisuje kolory kolejnych pasków dla podanej wartości rezystancji:

```
10 DIM B(3)
20 LET C$=' BLKBRONREDORGYELGRNBLUVLTGRAWHTGLD'
25 SCROLL
30 PRINT 'OHMS? ' ;
40 INPUT X
50 PRINT X
51 SCROLL
52 IF X  >=1 THEN GOTO 55
53 PRINT 'TOO SMALL'
54 GOTO 25
55 LET F=0
57 IF X  >=10 THEN GOTO 75
58 LET X =X*10
60 LET F=1
75 LET X=INT X
80 LET B(1)=VAL((STR$ X) (1))
100 LET B(2)=VAL((STR$ X) (2))
120 LET B(3)=(LEN(STR$ X))—2
121 IF F=1 THEN LET B(3)=10
130 FOR I=1 TO 3
135 LET B(I)=B(I)* 3+1
140 PRINT C$(B(I));C$(B(I)+1);C$(B(I)+2); ' ';
150 NEXT I
160 SCROLL
165 SCROLL
175 GOTO 25
```

Spostrzeżenia dotyczące możliwości ZX81 i powyższe procedury zaczerpnięto z rocznika 1982 czasopisma PERSONAL COMPUTER WORLD

AJP